

KODE MODUL

SWR.DEV.500.(1). A



**SEKOLAH MENENGAH KEJURUAN
BIDANG KEAHLIAN TEKNOLOGI INFORMASI DAN KOMUNIKASI
PROGRAM KEAHLIAN REKAYASA PERANGKAT LUNAK**

MENGUJI PROGRAM LEVEL 1



**BAGIAN PROYEK PENGEMBANGAN KURIKULUM
DIREKTORAT PENDIDIKAN MENENGAH KEJURUAN
DIREKTORAT JENDERAL PENDIDIKAN DASAR DAN MENENGAH
DEPARTEMEN PENDIDIKAN NASIONAL
2004**

KODE MODUL

SWR.DEV.500.(1). A



SEKOLAH MENENGAH KEJURUAN
BIDANG KEAHLIAN TEKNOLOGI INFORMASI DAN KOMUNIKASI
PROGRAM KEAHLIAN REKAYASA PERANGKAT LUNAK

Menguji Program Level 1



BAGIAN PROYEK PENGEMBANGAN KURIKULUM
DIREKTORAT PENDIDIKAN MENENGAH KEJURUAN
DIREKTORAT JENDERAL PENDIDIKAN DASAR DAN MENENGAH
DEPARTEMEN PENDIDIKAN NASIONAL
2004

KODE MODUL

SWR.DEV.500.(1). A



SEKOLAH MENENGAH KEJURUAN
BIDANG KEAHLIAN TEKNOLOGI INFORMASI DAN KOMUNIKASI
PROGRAM KEAHLIAN REKAYASA PERANGKAT LUNAK

Menguji Program Level 1

PENYUSUN
TIM FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA

BAGIAN PROYEK PENGEMBANGAN KURIKULUM
DIREKTORAT PENDIDIKAN MENENGAH KEJURUAN
DIREKTORAT JENDERAL PENDIDIKAN DASAR DAN MENENGAH
DEPARTEMEN PENDIDIKAN NASIONAL
2004

KATA PENGANTAR

Modul dengan judul “Menguji Program Level 1” merupakan bahan ajar yang digunakan sebagai panduan praktikum peserta diklat Sekolah Menengah Kejuruan (SMK) untuk membentuk salah satu bagian dari Kompetensi Menguji Program Level 1, Program Keahlian Rekayasa Perangkat Lunak.

Modul ini terdiri dari 4 (empat) kegiatan belajar. Kegiatan belajar 1 berisi operator pada Turbo Pascal. Kegiatan Belajar 2 berisi penggunaan statemen penyeleksian kondisi. Kegiatan Belajar 3 berisi tentang penggunaan pernyataan perulangan dan Kegiatan Belajar 4 berisi tentang pesan kesalahan kompilasi.

Modul ini terkait dengan modul lain yang membahas tentang pengoperasian sistem operasi, instalasi software, mengubah konfigurasi software, menyiapkan dan melakukan survey untuk menentukan kebutuhan data, mengoperasikan software bahasa pemrograman level 1, dan konversi data level1.

Yogyakarta, Desember 2004

Penyusun

Tim Fakultas Teknik

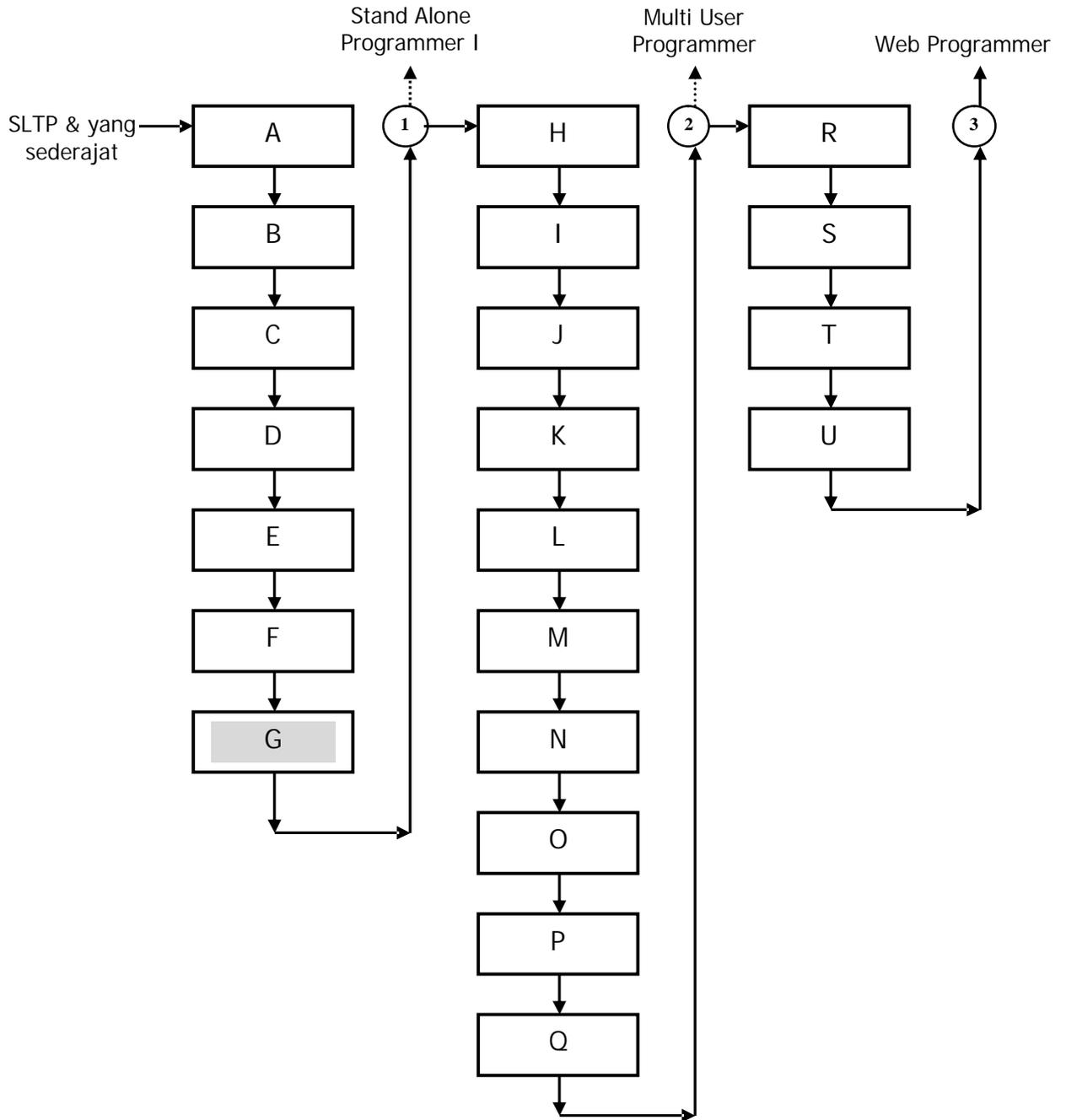
Universitas Negeri Yogyakarta

DAFTAR ISI MODUL

	Halaman
HALAMAN DEPAN	i
HALAMAN DALAM	ii
KATA PENGANTAR	iii
DAFTAR ISI	iv
PETA KEDUDUKAN MODUL	vi
PERISTILAHAN/ GLOSSARY	viii
I. PENDAHULUAN	1
A. DESKRIPSI JUDUL	1
B. PRASYARAT	1
C. PETUNJUK PENGGUNAAN MODUL	2
D. TUJUAN AKHIR.....	3
E. KOMPETENSI	3
F. CEK KEMAMPUAN	5
II. PEMELAJARAN	6
A. RENCANA PEMELAJARAN	6
B. KEGIATAN BELAJAR	7
1. Kegiatan Belajar 1: Operator Pada Turbo Pascal	7
a. Tujuan Kegiatan Pemelajaran	7
b. Uraian Materi 1	7
c. Rangkuman 1	13
d. Tugas 1	13
e. Tes Formatif 1	13
f. Kunci Jawaban Formatif 1	13
g. Lembar Kerja 1.....	14
2. Kegiatan Belajar 2 : Pernyataan Penyeleksian Kondisi	16
a. Tujuan Kegiatan Pemelajaran	16

b. Uraian Materi 2	16
c. Rangkuman 2	22
d. Tugas 2	22
e. Tes Formatif 2.....	22
f. Kunci Jawaban Formatif 2	23
g. Lembar Kerja 2.....	23
3. Kegiatan Belajar 3 : Pernyataan Perulangan	26
a. Tujuan Kegiatan Pemelajaran	26
b. Uraian Materi 3	26
c. Rangkuman 3	33
d. Tugas 3	33
e. Tes Formatif 3	33
f. Kunci Jawaban Formatif 3	33
g. Lembar Kerja 3	34
4. Kegiatan Belajar 4 : Pesan Kesalahan Kompilasi	36
a. Tujuan Kegiatan Pemelajaran	36
b. Uraian Materi 4	36
c. Rangkuman 4	57
d. Tugas 4	57
e. Tes Formatif 4	57
f. Kunci Jawaban Formatif 4	59
g. Lembar Kerja 4	61
III. EVALUASI	64
A. PERTANYAAN	64
B. KUNCI JAWABAN	64
C. KRITERIA PENILAIAN	66
IV.PENUTUP	67
DAFTAR PUSTAKA	68

PETA KEDUDUKAN MODUL



Keterangan :

Kode	Kode Kompetensi	Kompetensi
A	SWR.OPR.100.(1).A	Mengoperasikan Sistem Operasi
B	SWR.OPR.200.(1).A	Menginstalasi software
C	SWR.MNT.100.(1).A	Mengubah konfigurasi software
D	SWR.DEV.100.(1).A	Menyiapkan dan melakukan survey untuk menentukan kebutuhan data
E	SWR.OPR.309.(1).A	Mengoperasikan software bahasa pemograman level 1
F	DTA.OPR.115.(1).A	Konversi data level 1
G	SWR.DEV.500.(1).A	Menguji program level 1
H	HDW.OPR.103.(1).A	Mengoperasikan sistem operasi jaringan komputer berbasis teks
I	HDW.OPR.104.(1).A	Mengoperasikan sistem operasi jaringan komputer berbasis GUI
J	DTA.DEV.101.(3).A	Melakukan perancangan pengumpulan data
K	SWR.DEV.300.(2).A	Melakukan desain dan perancangan software
L	SWR.DEV.400.(2).A	Melakukan pengkodean program
M	DTA.MNT.101.(2).A	Melakukan back up data
N	DTA.MNT.102.(2).A	Melakukan restore data
O	SWR.OPR.303.(2).A	Mengoperasikan software aplikasi basis data
P	DTA.OPR.119.(2).A	Membuat query data
Q	SWR.DEV.500.(2).A	Menguji program
R	SWR.DEV.401.(2).A	Membangun interface dengan bahasa pemograman berorientasi objek
S	SWR.DEV.402.(3).A	Melakukan pengkodean program
T	SWR.OPR.304.(3).A	Mengoperasikan bahasa pemograman berbasis web
U	SWR.DEV.403.(2).A	Membangun program aplikasi remote data interaktif

PERISTILAHAN/ GLOSSARY

- Operator : simbol atau kata yang digunakan dalam program untuk melakukan suatu operasi, seperti jumlah atau perkalian, memberikan nilai ke variabel, membandingkan kesamaan dua buah nilai
- Ekspresi : Ungkapan.
- Clrscr : kepanjangan clear screen, yang artinya perintah untuk menghapus layar
- Eksekusi : istilah pemrograman di mana program baru dijalankan oleh komputer

BAB I

PENDAHULUAN

A. DESKRIPSI JUDUL

Menguji Program Level 1 merupakan modul praktikum berisi tentang tanda operasi, pernyataan penyeleksian kondisi dan pernyataan perulangan. Modul ini terdiri dari 4 (empat) kegiatan belajar. Kegiatan belajar 1 berisi operator pada bahasa pemrograman Turbo Pascal. Turbo Pascal dipilih dalam modul ini karena strukturnya yang tegas di samping tidak menuntut persyaratan hardware yang canggih. Kegiatan Belajar 2 berisi penggunaan statemen penyeleksian kondisi. Kegiatan Belajar 3 berisi tentang penggunaan pernyataan perulangan dan Kegiatan Belajar 4 berisi tentang pesan kesalahan kompilasi

Setelah menguasai modul ini diharapkan peserta diklat mampu menguji program level 1, yang meliputi penggunaan operasi, pernyataan penyeleksian kondisi, pernyataan perulangan dalam pemrograman Pascal. Modul ini terkait dengan modul lain yang membahas tentang pengoperasian sistem operasi, instalasi software, mengubah konfigurasi software, menyiapkan dan melakukan survey untuk menentukan kebutuhan data, mengoperasikan software bahasa pemrograman level 1 dan konversi data level 1.

B. PRASYARAT

Modul Menguji Program Level 1 memerlukan persyaratan yang harus dimiliki oleh peserta diklat, yaitu telah lulus modul Mengoperasikan PC Stand Alone dengan sistem operasi berbasis GUI dan modul Mengoperasikan Periferal.

C. PETUNJUK PENGGUNAAN MODUL

1. Petunjuk bagi Peserta Diklat

Peserta diklat diharapkan dapat berperan aktif dan berinteraksi dengan sumber belajar yang dapat digunakan, karena itu harus memperhatikan hal-hal sebagai berikut :

a. Langkah-langkah belajar yang ditempuh

- 1) Persiapkan alat dan bahan !
- 2) Bacalah dengan seksama uraian materi pada setiap kegiatan belajar !
- 3) Cermatilah langkah langkah kerja pada setiap kegiatan belajar sebelum mengerjakan, bila belum jelas tanyakan pada instruktur !
- 4) Kembalikan semua peralatan praktik yang digunakan!

b. Perlengkapan yang Harus Dipersiapkan

Guna menunjang keselamatan dan kelancaran tugas/ pekerjaan yang harus dilakukan, maka persiapkanlah seluruh perlengkapan yang diperlukan, pelajarilah terlebih dahulu modul ini dan buku-buku yang menunjang.

c. Hasil Pelatihan

Peserta diklat mampu melakukan tugas :

- 1) Memahami Operator Pada Turbo Pascal
- 2) Menggunakan pernyataan penyeleksian kondisi
- 3) Menggunakan statemen perulangan
- 4) Memahami pesan kesalahan kompilasi

2. Peran Guru

Guru yang akan mengajarkan modul ini hendaknya mempersiapkan diri sebaik-baiknya yaitu mencakup aspek strategi pembelajaran, penguasaan materi, pemilihan metode, alat bantu media pembelajaran dan perangkat evaluasi.

Guru harus menyiapkan rancangan strategi pembelajaran yang mampu mewujudkan peserta diklat terlibat aktif dalam proses pencapaian/penguasaan kompetensi yang telah diprogramkan. Penyusunan rancangan strategi pembelajaran mengacu pada kriteria unjuk kerja (KUK) pada setiap sub kompetensi yang ada dalam GBPP.

D. TUJUAN AKHIR

Peserta diklat dapat menguji program level 1 yang meliputi penggunaan tanda operasi, penyeleksian kondisi dan pengulangan proses dalam pemrograman Pascal dengan benar.

E. KOMPETENSI

Sub Kompetensi	Kriteria Unjuk Kerja	Lingkup Belajar	Materi Pokok Pembelajaran		
			Sikap	Pengetahuan	Ketrampilan
1	2	3	4	5	6
1. Mempersiapkan program yang akan diuji	<ul style="list-style-type: none">• Perlengkapan kerja diidentifikasi sesuai dengan kebutuhan• User manual program yang akan diuji tersedia	<ul style="list-style-type: none">• Validasi software aplikasi• Struktur alur pemrograman	Mengikuti SOP dalam mempersiapkan langkah-langkah pengujian software	<ul style="list-style-type: none">• Memahami user manual program aplikasi yang akan diuji• Menentukan prosedur kerja pengujian program aplikasi yang akan diuji (SOP)	<ul style="list-style-type: none">• Mempersiapkan peralatan yang akan digunakan untuk menguji software

<p>2. Melakukan pengujian program</p>	<ul style="list-style-type: none"> • Proses pengujian program aplikasi dilakukan sesuai dengan SOP • Penilaian terhadap fungsi-fungsi program aplikasi yang dapat maupun tidak dapat dijalankan 	<ul style="list-style-type: none"> • Validasi software aplikasi • Struktur alur pemrograman 	<p>Mengikuti SOP dalam melaksanakan langkah-langkah pengujian software</p>	<ul style="list-style-type: none"> • Membuktikan kebenaran/kesahihan fungsi-fungsi alur pemrograman dari software yang diuji 	<ul style="list-style-type: none"> • Mempraktikkan langkah-langkah pengujian software
<p>3. Melaporkan hasil pengujian</p>	<ul style="list-style-type: none"> • Laporan dibuat sesuai dengan format dan prosedur/instruksi kerja yang ditetapkan • Laporan diserahkan kepada pembuat program aplikasi untuk digunakan atau diperbaiki 	<ul style="list-style-type: none"> • Validasi software aplikasi • Struktur alur pemrograman 	<p>Mengikuti SOP dalam melaksanakan langkah-langkah pengujian software</p>	<ul style="list-style-type: none"> • Menunjukkan hasil pengujian software 	<ul style="list-style-type: none"> • Mengidentifikasi hasil pengujian software

F. CEK KEMAMPUAN

Isilah cek list (√) seperti pada tabel di bawah ini dengan sikap jujur dan dapat dipertanggung jawabkan untuk mengetahui kemampuan awal yang telah dimiliki.

Kompetensi	Pernyataan	Saya dapat Melakukan Pekerjaan ini dengan Kompeten		Bila Jawaban "Ya" Kerjakan
		Ya	Tidak	
Menguji Program Level 1	1. Memahami Operator Pada Turbo Pascal			Tes Formatif 1
	2. Menggunakan statemen penyeleksian kondisi			Tes Formatif 2
	3. Menggunakan statemen perulangan			Tes Formatif 3
	4. Memahami pesan kesalahan kompilasi			Tes Formatif 4

Apabila anda menjawab TIDAK pada salah satu pernyataan di atas, maka pelajarilah modul ini.

BAB II PEMELAJARAN

A. RENCANA PEMELAJARAN

Kompetensi : Menguji Program Level 1

Jenis Kegiatan	Tanggal	Waktu	Tempat Belajar	Alasan Perubahan	Tanda Tangan Guru
Memahami operator pada Turbo Pascal					
Menggunakan statement penyeleksian kondisi					
Menggunakan statement perulangan					
Memahami pesan kesalahan kompilasi					

B. KEGIATAN BELAJAR

1. Kegiatan Belajar 1: Operator pada Turbo Pascal

a. Tujuan Kegiatan Pemelajaran

Setelah melaksanakan kegiatan pemelajaran 1 diharapkan peserta diklat dapat memahami operator pada Turbo Pascal.

b. Uraian Materi 1

Operator yang dibahas pada uraian materi di bawah ini, terdapat 3 jenis, yaitu matematika, relasi, boolean.

1) Operator dan Operasi Matematika

a) Operator Matematika

Operator adalah simbol atau kata yang digunakan dalam program untuk melakukan suatu operasi, seperti jumlah atau perkalian, memberikan nilai ke variabel, membandingkan kesamaan dua buah nilai. Nilai yang dioperasikan oleh operator bersama operand membentuk suatu ekspresi/ ungkapan. Kalau dalam suatu ekspresi terdapat lebih dari satu operator, maka perhitungannya ditentukan berdasarkan urutan prioritas, misal:

$1+2*3$ akan memberikan hasil 7. hasil 7 diperoleh karena $2*3$ dikerjakan terlebih dahulu, mengingat urutan pengerjaan * lebih tinggi daripada plus. Urutan prioritas dapat dilihat pada tabel berikut ini:

Operator	Arti	Prioritas	Contoh
*	Kali	1	$2*3=6$
/	Bagi (real)	1	$7/2=3.5$
DIV	Bagi (bulat)	1	$7 \text{ div } 2=3$
MOD	Sisa pembagian	1	$7 \text{ mod } 2 =1$
+	Tambah	2	$2+3=5$
-	Kurang	2	$2-3= -1$

Operasi yang tidak dapat dilaksanakan dengan operator yang tersedia dapat dilaksanakan dengan menggunakan fungsi-fungsi yang disediakan Turbo pascal.

Contoh : Operasi Penjumlahan dua buah bilangan

```

program jumlah;
uses crt;
var    a,b,c :integer;
begin
    write('masukkan data pertama : ');readln(a);
    write('masukkan data kedua  : ');readln(b);
    c:=a+b;
    writeln('hasil penjumlahan : ',c);readln;
end.

```

b) Pernyataan Inc dan Dec

Bentuk pernyataan Inc :

```

Inc(Variabel)
Inc(Variabel, Nilai)

```

Parameter Nilai menyatakan besarnya nilai yang akan ditambahkan terhadap isi semula dari variabel.

Contoh:

$x:=x+1$; dapat dituliskan $\text{inc}(x)$;

$x:=x+2$; dapat dituliskan $\text{inc}(x,2)$;

Bentuk pernyataan Dec:

$\text{Dec}(\text{Variabel})$

$\text{Dec}(\text{Variabel}, \text{Nilai})$

Parameter Nilai menyatakan besarnya nilai untuk mengurangi terhadap isi semula dari variabel.

Contoh :

$x:=x-1$; dapat dituliskan $\text{dec}(x)$;

$x:=x-2$; dapat dituliskan $\text{dec}(x,2)$;

c) Fungsi Matematika

Tabel Fungsi Matematika, adalah sebagai berikut:

Fungsi	Keterangan
Abs(x)	Digunakan untuk memperoleh nilai mutlak (absolut) x. Dalam hal ini tipe hasil sesuai dengan parameter
ArcTan(x)	Untuk memperoleh arstangent dari nilai x. Hasilnya berupa sudut yang dinyatakan dalam satuan radian. Baik parameter maupun hasilnya bertipe real
Cos(x)	Untuk memperoleh nilai cosinus dari sudut x. Sudut dinyatakan dalam radian. Parameter dan hasilnya bertipe Real
Exp(x)	Menghasilkan nilai eksponesial dari $x(e^x)$. Baik parameter maupun hasilnya bertipe real
Frac(x)	Memberikan bagian pecahan dari argumen x. Bai parameter maupun hasilnya bertipe real
Int(x)	Mendapatkan bagian bulat dari argumen x. Hasilnya bertipe real
Ln(x)	Menghasilkan logaritma alami dari $x(e^{\log x})$. Baik parameter maupun hasilnya bertipe real
Pi	Menghasilkan nilai π (3,14159265356)
Round(x)	Melakukan pembulatan terhadap parameter bertipe real ke bilangan bulat terdekat. Hasilnya bertipe longint
Sin(x)	Untuk memperoleh nilai sinus dari sudut x (dalam radian)
Sqr(x)	Untuk memperoleh nilai kuadrat dari x. Tipe hasilnya sesuai dari tipe parameter
Sqrt(x)	Untuk memperoleh akar kuadrat dari x. Hasil dan parameternya

	bertipe Real
Trunc(x)	Memperoleh nilai bulat dari parameter bertipe Real. Tipe hasilnya berupa LongInt

Fungsi-fungsi standar yang berkaitan dengan bilangan dapat dilihat pada tabel berikut ini :

Fungsi	Keterangan
Odd(bil)	Fungsi ini memberikan hasil berupa True kalau bil (bertipe LongInt berupa bilangan ganjil)
Random(bil)	Fungsi ini memberikan hasil berupa bilangan acak (bertipe Word) yang nilainya lebih besar atau sama dengan nol tetapi kurang dari bil. Dalam hal ini, bil bertipe Word
Random	Fungsi ini memberikan hasil berupa bilangan acak yang lebih besar atau sama dengan nol, tetapi kurang dari 1

2) Operator Relasi

Operator relasi (operator hubungan) biasa dipakai untuk melakukan perbandingan. Seluruh operator relasi dapat dilihat sebagai berikut:

Operator	Keterangan
>	Lebih dari
<	Kurang dari
>=	Lebih dari atau sama dengan
<=	Kurang dari atau sama dengan
<>	Tidak sama dengan
=	Sama dengan
IN	Perbandingan suatu nilai dengan suatu himpunan

Operator relasi dapat dikenakan pada berbagai jenis data, misalnya Integer, Byte, Char, Boolean dan String.

Bentuk perbandingan :

Operand1 Operator Operand2

Contoh

Ekspresi $2 < 7$, hasilnya True

Hasil perbandingan berupa True atau False.

Operator Boolean

Operator boolean bermanfaat untuk melakukan operasi nalar "dan", "tidak/bukan", "atau" ataupun "atau eksklusif". operator untuk operasi ini diantaranya adalah AND (operasi "dan"), OR (operasi "atau"), XOR (operasi "atau eksklusif") dan NOT (operasi "tidak/bukan"). Masing-masing dijelaskan sebagai berikut :

a) Operator AND

Operator AND digunakan untuk membandingkan dua buah elemen, hasilnya akan benar bila keduanya benar. Bila logika benar diberi simbol 1 dan logika salah diberi simbol 0, maka hubungan operator AND tampak pada tabel berikut ini:

A	B	A AND B
1	1	1
1	0	0
0	1	0
0	0	0

Contoh penggunaan AND adalah sebagai berikut :

$(\text{pil} \geq 1) \text{ AND } (\text{pil} \leq 8)$

ekspresi diatas menghasilkan nilai True sekiranya isi Pil berkisar antara 1 sampai dengan 8. bila Pil selain angka tersebut maka akan bernilai False.

b) Operator OR

Operator OR digunakan untuk membandingkan dua buah elemen, hasilnya akan benar bila salah satu atau keduanya benar. Bila logika benar diberi simbol 1 dan logika salah diberi simbol 0, maka hubungan kedua operator OR tampak pada tabel berikut ini:

A	B	A OR B
1	1	1
1	0	1
0	1	1
0	0	0

Contoh pemakaian OR :

(jawab='A') OR (jawab='a')

Hasil dari ekspresi diatas adalah akan menghasilkan nilai True apabila di jawab A dan a, dan akan bernilai salah apabila dijawab selain huruf tersebut.

c) Operator XOR

Operator XOR digunakan untuk membandingkan dua buah elemen, hasilnya akan benar bila salah satunya yang benar. Bila logika benar diberi simbol 1 dan logika salah diberi simbol 0, maka hubungan operator XOR tampak pada tabel berikut ini:

A	B	A XOR B
1	1	0
1	0	1
0	1	1
0	0	0

d) Operator NOT

NOT hanya melibatkan sebuah operand, dengan bentuk ekspresi berupa:

NOT operand

Hasil ekspresi adalah:

True, kalau operand False

False, kalau operand True.

c. Rangkuman 1

Operator pada Turbo Pascal terdapat bermacam-macam, antara lain operator matematika, operator relasi dan operator boolean. operator matematis Operator boolean masih dibagi lagi menjadi operator And, operator Or, operator XOR dan operator NOT.

d. Tugas 1

- 1) Pelajarilah uraian materi tentang Operator pada Turbo Pascal !
- 2) Hitunglah
 - a) $56 \bmod 5$
 - b) $64 \div 3$
- 3) Buatlah ekspresi dalam Pascal untuk menyatakan Y terletak antara 2 dan 5!

e. Tes Formatif 1

- 1) Sebutkan fungsi dari operator relasi ?
- 2) Sebutkan kegunaan operator AND ?
- 3) Buatlah ekspresi dalam Pascal untuk menyatakan :

a) $\sqrt{(a+b)/c}$

b) $\frac{y^3}{2} - 5y + 2$

f. Kunci Jawaban Formatif 1

- 1) Operator relasi (operator hubungan) biasa dipakai untuk melakukan perbandingan.
- 2) Operator AND digunakan untuk membandingkan dua buah elemen, hasilnya akan benar bila keduanya benar
- 3) Ekspresi Turbo Pascal

a) $\text{Sqrt}((a+b) / c)$

b) $y*y*y / 2-5*y+2$

g. Lembar Kerja 1

Alat dan Bahan

PC yang sudah terinstal Turbo Pascal 7.0

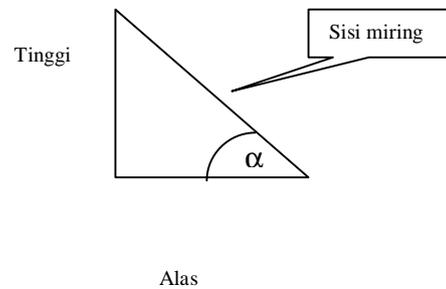
Kesehatan dan Keselamatan Kerja

- 1) Berdo'alah sebelum memulai kegiatan belajar.
- 2) Bacalah dan pahami petunjuk praktikum pada setiap lembar kegiatan belajar.
- 3) Pastikan komputer dalam keadaan baik, semua kabel penghubung terhubung dengan benar.
- 4) Jangan meletakkan benda yang dapat mengeluarkan medan elektromagnetik di dekat komputer (magnet, handphone, dan sebagainya).
- 5) Gunakanlah komputer sesuai fungsinya dengan hati-hati, jangan bermain game.
- 6) Setelah selesai, matikan komputer dengan benar!

Langkah Kerja

- 1) Siapkanlah peralatan yang akan digunakan!
- 2) Periksa semua kabel penghubung pada komputer.
- 3) Hidupkan komputer dengan menekan saklar pada komputer, jangan menghidupkan komputer dengan memasukkan colokan ke stop kontak ketika saklar dalam keadaan on.
- 4) Jalankan program Turbo Pascal.
- 5) Buatlah program yang meminta data alas dan tinggi suatu segitiga siku-siku dimasukkan dari keyboard. Kemudian program

melaporkan panjang sisi miring dan sudut antara sisi miring dan alas dalam satuan derajat.



Gambar 1. Gambar Soal 5)

- 6) Ujilah program anda !
- 7) Setelah selesai, laporkan hasil kerja Anda kepada instruktur.
- 8) Setelah diteliti matikan komputer dan rapikan seperti semula.

2. Kegiatan Belajar 2 : Pernyataan Penyeleksian Kondisi

a. Tujuan Kegiatan Pemelajaran

Setelah melaksanakan kegiatan pemelajaran 2 diharapkan peserta diklat dapat menggunakan pernyataan penyeleksian kondisi .

b. Uraian Materi 2 : Penyeleksian Kondisi

Turbo Pascal menyediakan dua macam pernyataan untuk mengambil keputusan, yaitu:

1) Pernyataan IF

Pernyataan IF mempunyai bermacam-macam bentuk, yaitu:

a) Pernyataan IF sederhana

Bentuk dari pernyataan IF yang sederhana adalah sebagai berikut:

```
IF kondisi THEN pernyataan
```

Pada bentuk ini, pernyataan hanya akan dijalankan kalau kondisi bernilai TRUE.

Contoh program IF:

```
Program contoh_if1;  
Uses crt;  
Var nilai: integer;  
Begin  
Clrscr;  
Write('Ketikkan Nilai Anda : ');readln(nilai);  
If nilai>60 then  
    Writeln('Lulus'); readln;  
End.
```

b) Pernyataan IF ELSE

Bentuk IF ELSE adalah sebagai berikut :

```
IF kondisi THEN
Pernyataan1
ELSE
Pernyataan2;
```

Pada bentuk ini apabila pernyataan1 tidak terpenuhi maka akan menjalankan pernyataan ke dua.

```
Program contoh_if2;
Uses crt;
Var nilai: integer;
Begin
Clrscr;
Write('Ketikkan Nilai Anda : ');readln(nilai);
If nilai>60 then
    Writeln('Lulus')
Else
    Writeln('Gagal');readln;
End.
```

Pemakaian operator AND pada IF :

```
Program contoh_if3;
Uses crt;
Var pil: integer;
Begin
Clrscr;
Write('Ketikkan pilihan [1...4] setelah itu tekan enter:
');readln(pil);
```

```

If (pil>=1) and (pil<=4) then
    Writeln('pilihan benar')
Else
    Writeln('pilihan salah');readln;
End.

```

c) Pernyataan IF bersarang

Pernyataan IF bersarang adalah dimana pada pernyataan IF mengandung pernyataan IF yang lainnya. Suatu pedoman untuk memahami pernyataan IF ELSE adalah sebagai berikut :

- (1) ELSE berpasangan dngan IF yang tepat di atasnya, yang terletak dalam satu blok
- (2) Bila IF yang terletak tepat sebelum ELSE terletak di dalam BEGIN END, IF dikatakan tidak terletak dalam satu blok yang sama.

Contoh bentuk IF bersarang :

```

IF kondisi1 THEN
    IF kondisi2 THEN
        Pernyataan_1
    ELSE
        Pernyataan_2;

```

Pada bentuk tersebut ELSE berpasangan dengan kondisi2

Contoh yang lainnya:

```

IF kondisi1 THEN
    begin
        IF kondisi2 THEN
            Pernyataan_1;
    end

```

ELSE

Pernyataan_2;

Pada contoh tersebut ELSE berpasangan dengan kondisi 1.

Contoh program menggunakan IF bersarang adalah sebagai berikut:

Kriteria	Nilai Huruf
$\text{Skore} \geq 85$	A
$76 \leq \text{skore} < 85$	B
$66 \leq \text{skore} < 76$	C
$56 \leq \text{skore} < 66$	D
$\text{Skore} < 56$	E

Program nilai;

```
uses crt;
```

```
var skore :integer;
```

```
begin
```

```
clrscr;
```

```
write('Masukan nilai :');readln(skore);
```

```
if skore >= 85 then
```

```
    writeln('A')
```

```
else
```

```
    if skore >= 76 then
```

```
        writeln('B')
```

```
    else
```

```
        if skore >= 66 then
```

```
            writeln('C')
```

```
        else
```

```
            if skore >= 56 then
```

```
                writeln('D')
```

```

else
    writeln('E');readln;
end.

```

2) Pernyataan CASE

Pernyataan CASE merupakan alternatif dari pernyataan IF untuk masalah dengan pilihan berganda. Pada masalah tertentu CASE lebih memberikan kejelasan daripada IF. Namun semua persoalan yang dapat ditangani oleh CASE bisa ditangani oleh IF, tetapi sebaliknya.

Format pernyataan CASE OF adalah sebagai berikut :

```

CASE nilai OF
    Nilai_1: pernyataan_1;
    Nilai_2: pernyataan_2;
    .....
    Nilai_k: pernyataan_k;
END;

```

Contoh program Nilai ujian mahasiswa dalam bentuk A (berarti Sangat Baik), B (berarti Baik), C (berarti Cukup), D (berarti Kurang) dan E (berarti Gagal).

```

Program contoh_case1;
Uses crt;
Var nilai:char;
Begin
Clrscr;
Write('Tuliskan Nilai huruf yang di dapat dengan huruf Kapital
: ');readln(nilai);
Case nilai of
    'A': writeln('Sangat Baik');

```

```

        'B': writeln('Baik');
        'C': writeln('Cukup');
        'D': writeln('Kurang');
        'E': writeln('Gagal');
    end;readln;
end.

```

Struktur CASE OF...ELSE merupakan pengembangan dari struktur CASE OF. Pada struktur CASE OF apabila ada kondisi yang tidak terpenuhi berarti tidak ada pernyataan di dalam lingkungan CASE OF yang diproses. Struktur CASE OF... ELSE, bila ada kondisi tidak terpenuhi maka pernyataan yang akan diproses adalah pernyataan yang ada di ELSE.

Contoh penggunaan CASE OF... ELSE adalah sebagai berikut:

```

Program contoh_case2;
Uses crt;
Var nilai:integer;
Begin
Clrscr;
Write('Pilihlah angka[1,2,3] : ');readln(nilai);
Case nilai of
    1: writeln('Dapat Kulkas');
    2: writeln('Dapat TV');
    3: writeln('dapat Mobil');
else
    writeln('Pilihan Salah');
end;readln;
end.

```

c. Rangkuman 2

Bahasa Pascal menyediakan pernyataan untuk menyeleksi suatu kondisi, yaitu IF dan CASE. Dengan menyeleksi suatu kondisi, program dapat menentukan tindakan apa yang harus dikerjakan tergantung dari kondisi yang diseleksi tersebut.

d. Tugas 2

- 1) Pelajarilah uraian materi tentang Penyeleksian Kondisi!
- 2) Buatlah program yang meminta agar sebuah bilangan bulat dimasukkan dari keyboard dan kemudian program dapat mengomentari bilangan tersebut genap atau ganjil.

e. Tes Formatif 2

- 1) Apakah yang salah dari penulisan IF berikut :

```
IF bil<0 then
  Writeln('Negatif');
Else
  Writeln('Positif');
```

- 2) Ujilah program di bawah ini dan sebutkan kesalahannya ?

```
Program tes suhu;
Uses crt;
Var suhu: integer;
Begin
Clrscr;
Write('Ketikkan Suhu : ');readln(suhu);
  If nilai>37 then
    Writeln('suhu rendah');
  Else
```

```
        Writeln('suhu tinggi');  
    End;
```

f. Kunci Jawaban Formatif 2

- 1) Penulisan yang salah pada pernyataan IF tersebut adalah di depan else tidak boleh ada koma.

```
    IF bil<0 then  
        Writeln('Negatif')  
    Else  
        Writeln('Positif');
```

- 2) Pembeneran Program :

```
Program tes_suhu;  
Uses crt;  
Var suhu: integer;  
Begin  
Clrscr;  
Write('Ketikkan Suhu : ');readln(suhu);  
    If nilai>37 then  
        Writeln('suhu rendah')  
    Else  
        Writeln('suhu tinggi');readln;  
End.
```

g. Lembar Kerja 2

Alat dan Bahan

PC yang sudah terinstal Turbo Pascal 7.0

Kesehatan dan Keselamatan Kerja

- 1) Berdo'alah sebelum memulai kegiatan belajar.
- 2) Bacalah dan pahami petunjuk praktikum pada setiap lembar kegiatan belajar.
- 3) Pastikan komputer dalam keadaan baik, semua kabel penghubung terhubung dengan benar.
- 4) Jangan meletakkan benda yang dapat mengeluarkan medan elektromagnetik di dekat komputer (magnet, handphone, dan sebagainya).
- 5) Gunakanlah komputer sesuai fungsinya dengan hati-hati, jangan bermain game.
- 6) Setelah selesai, matikan komputer dengan benar!

Langkah Kerja

- 1) Siapkanlah peralatan yang akan digunakan!
- 2) Periksa semua kabel penghubung pada komputer.
- 3) Hidupkan komputer dengan menekan saklar pada komputer, jangan menghidupkan komputer dengan memasukkan colokan ke stop kontak ketika saklar dalam keadaan on.
- 4) Jalankan program Turbo Pascal.
- 5) Ujilah setiap contoh program pada uraian materi 2 di atas.
- 6) Buatlah program yang meminta sebuah bilangan dimasukkan dari keyboard. Kemudian program dijalankan memberikan komentar
 - a) Negatif kalau nilai kurang dari nol
 - b) Positif kalau nilai lebih dari nol
 - c) Nol kalau nilai samadengan nol

- 7) Ujilah program anda !
- 8) Setelah selesai, laporkan hasil kerja Anda kepada instruktur.
- 9) Setelah diteliti matikan komputer dan rapikan seperti semula.

3. Kegiatan Belajar 3 : Pernyataan Perulangan

a. Tujuan Kegiatan Pemelajaran

Setelah melaksanakan kegiatan pemelajaran 3 diharapkan peserta diklat dapat memahami dan menggunakan pernyataan perulangan Turbo Pascal

b. Uraian Materi 3

Turbo Pascal menyediakan 3 macam pernyataan, yaitu FOR, WHILE dan REPEAT.

1) Pernyataan FOR

Pernyataan FOR digunakan untuk melakukan pengulangan yang jumlahnya telah diketahui sebelumnya. Berikut ini akan diterangkan macam-macam bentuk FOR.

a) FOR TO (disebut juga perulangan positif)

Perulangan FOR TO merupakan perulangan dengan penghitung dari kecil ke besar dan bentuknya adalah sebagai berikut :

FOR variabel:=nilai_awal TO nilai_akhir DO pernyataan

Syarat dalam bentuk FOR ini adalah

- (1) Variabel, nilai_awal dan nilai_akhir harus bertipe sama
- (2) Tipe yang digunakan pada ketiga komponen tersebut harus berupa ordinal
- (3) Nilai_awal harus lebih kecil atau sama dengan nilai_akhir. Kalau nilai_akhir lebih kecil dari nilai_awal maka bagian pernyataan tidak dapat dijalankan.

Contoh pemakaian FOR TO untuk pengulangan tulisan Pascal sebanyak 3 kali adalah sebagai berikut:

```
program cobafor1;
uses crt;
var ulang: integer;
begin
clrscr;
FOR ulang:=1 TO 3 DO
    writeln('Pascal');readln;
End.
```

Apabila dijalankan program tersebut akan menampilkan tulisan Pascal urut ke bawah sebanyak 3 kali.

Contoh Program menampilkan huruf A sampai T, secara urut ke kanan.

```
program cobafor2;
uses crt;
var ulang: char;
begin
clrscr;
FOR ulang:='A' TO 'T' DO
    write(ulang);readln;
End.
```

- b) FOR DOWNTO (disebut juga perulangan negatif)

Perulangan FOR DOWNTO merupakan perulangan dengan penghitungan dari besar ke kecil dan bentuknya adalah sebagai berikut:

FOR variabel:=nilai_awal DOWNTO nilai_akhir DO pernyataan

Syarat dalam bentuk FOR ini adalah

- (1) Variabel, nilai_awal dan nilai_akhir harus bertipe sama
- (2) Tipe yang digunakan pada ketiga komponen tersebut harus berupa ordinal
- (3) Nilai_awal harus lebih besar atau sama dengan nilai_akhir. Kalau nilai_akhir lebih besar dari nilai_awal maka bagian pernyataan tidak dapat dijalankan.

Contoh Program FOR DOWNTO untuk menampilkan huruf T sampai A, secara urut ke kanan.

```
program cobafor3;
uses crt;
var ulang: char;
begin
clrscr;
    FOR ulang:='T' DOWNTO 'A' DO
        write(ulang);readln;
    End.
```

c) FOR bersarang

Bentuk perulangan bersarang ini adalah perulangan yang berada di dalam perulangan yang lainnya. Perulangan yang lebih dalam akan diproses terlebih dahulu sampai habis, kemudian perulangan yang lebih luar baru dikerjakan.

```
program cobafor4;
uses crt;
var ulang1, ulang2: integer;
begin
clrscr;
    FOR ulang1:=1 TO 5 DO
```

```

Begin
FOR ulang2:=1 TO 3 Do
    Write(ulang1;ulang2);
    Writeln;
End; readln;
End.

```

2) Pernyataan While

Pernyataan While biasa dipakai untuk melakukan pengulangan yang jumlahnya tidak diketahui di depan atau selang pencacah tidak sebesar 1 atau -1, sehingga pernyataan akan dilakukan terus menerus selama kondisi bernilai True bentuk pernyataan While:

WHILE kondisi DO pernyataan

Contoh sebuah program untuk mengurutkan angka dari 10 sampai dengan 1, dengan penurunan sebanyak 1.

```

program cobawhile_1;
uses crt;
var ulang: integer;
begin
clrscr;
ulang:=10; {Nilai awal}
while ulang>=1 do
begin
writeln(ulang);
ulang:=ulang-1;
end;
readln;
End.

```

Contoh potongan program untuk apabila kenaikan atau penurunan tidak sebesar 1.

```
Ulang:=1;  
WHILE ulang <= 10 DO  
Begin  
  Write(ulang);  
  Ulang:=ulang+2;  
End;
```

Bagian yang diulang oleh While juga bisa mengandung WHILE yang lainnya, inilah yang disebut dengan WHILE bersarang.

Contoh program WHILE bersarang adalah sebagai berikut :

```
program cobawhile2;  
uses crt;  
var nilai, total, rata:real;  
    N,I : integer;  
    lagi: char;  
  
begin  
  lagi:='Y';  
  while lagi='Y' do  
    begin  
      clrscr;  
      write('Jumlah data :'); readln(n);  
      writeln;  
  
      i:=0;  
      total:=0;  
      while i<n do
```

```

begin
  i:=i+1;
  write('Nilai data ke',i, ':'); readln (nilai);
  total:=total + nilai;
end;
rata:=total/n;

writeln;
writeln('Total nilai    = ', total:8:2);
writeln('Jumlah data    = ', n:3);
writeln('Rata-rata nilai = ', rata:8:2);
writeln;
writeln('Hitung lagi (Y/T) ');readln(lagi);
end;
end.

```

3) Pernyataan Repeat....Until

Repeat...Until digunakan untuk mengulang (repeat) pernyataan sampai (until) kondisi diseleksi di Until tidak terpenuhi. Persamaan dengan while adalah dipakai untuk menangani pengulangan yang jumlahnya belum pasti.

Perbedaan antara Repeat..Until dengan While adalah

- a) Paling sedikit pernyataan didalam perulangan Repeat..Until diproses sekali, karena seleksi kondisi ada pada pernyataan Until yang terletak di bawah. Sedang pada struktur While paling sedikit dikerjakan nol kali, karena kondisi ada pada pernyataan while yang terletak di atas, sehingga kalau kondisi sudah tidak terpenuhi maka tidak akan masuk ke dalam lingkungan perulangannya.

- b) Pada Repeat..until pada pernyataan boleh tidak menggunakan Begin End untuk menunjukkan batas perulangan karena batas perulangan sudah ditunjukkan oleh Repeat sampai dengan Until

Contoh program menggunakan Repeat...Until adalah sebagai berikut:

```
Program contoh_repeat1;
Uses crt;
Var ulang : integer;
Begin
  Ulang:= 0;
  Repeat
    Ulang:=ulang+1;
    Writeln(ulang);
  Until ulang=5; readln;
End.
```

Contoh program repeat menggunakan operator or

```
Program contoh_repeat2;
Uses crt;
Var ulang : char;
Begin
  Repeat
    Writeln('Pascal');
    Writeln('Diulang sampai ditekan huruf t');
    readln(ulang);
  Until (ulang='T') OR (ulang='t');
End.
```

c. Rangkuman 3

Turbo Pascal menyediakan 3 macam bentuk perulangan, yaitu FOR, WHILE dan REPEAT...UNTIL. Ketiga bentuk tersebut masing-masing masih dibagi lagi dari bentuk yang paling sederhana sampai ke bentuk bersarang.

d. Tugas 3

- 1) Pelajarilah uraian materi tentang Penggunaan Pernyataan Perulangan !
- 2) Apakah yang dimaksud dengan perulangan positif dan negatif?
- 3) Buatlah program untuk menampilkan 30 buah tanda * pada satu baris dengan menggunakan perulangan FOR, WHILE dan REPEAT...UNTIL!

e. Tes Formatif 3

- 1) Sebutkan persamaan dan perbedaan antara WHILE dan REPEAT...UNTIL ?
- 2) Sebutkan kegunaan pernyataan perulangan FOR ?

f. Kunci Jawaban Formatif 3

- 1) Persamaan antara WHILE dan REPEAT...UNTIL adalah adalah dipakai untuk menangani pengulangan yang jumlahnya belum pasti

Perbedaan antara Repeat..Until dengan While adalah

- a) Paling sedikit pernyataan didalam perulangan Repeat..Until diproses sekali, karena seleksi kondisi ada pada pernyataan Until yang terletak di bawah. Sedang pada struktur While paling sedikit dikerjakan nol kali, karena kondisi ada pada pernyataan while yang terletak di atas, sehingga kalau

kondisi sudah tidak terpenuhi maka tidak akan masuk ke dalam lingkungan perulangannya.

b) Pada Repeat..until pada pernyataan boleh tidak menggunakan Begin End untuk menunjukkan batas perulangan karena batas perulangan sudah ditunjukkan oleh Repeat sampai dengan Until

2) Kegunaan pernyataan FOR adalah untuk melakukan pengulangan yang jumlahnya telah diketahui sebelumnya.

g. Lembar Kerja 3

Alat dan Bahan

PC yang sudah terinstal Turbo Pascal 7.0

Kesehatan dan Keselamatan Kerja

- 1) Berdo'alah sebelum memulai kegiatan belajar.
- 2) Bacalah dan pahami petunjuk praktikum pada setiap lembar kegiatan belajar.
- 3) Pastikan komputer dalam keadaan baik, semua kabel penghubung terhubung dengan benar.
- 4) Jangan meletakkan benda yang dapat mengeluarkan medan elektromagnetik di dekat komputer (magnet, handphone, dan sebagainya).
- 5) Gunakanlah komputer sesuai fungsinya dengan hati-hati, jangan bermain game.
- 6) Setelah selesai, matikan komputer dengan benar!

Langkah Kerja

- 1) Siapkanlah peralatan yang akan digunakan!
- 2) Periksa semua kabel penghubung pada komputer.

- 3) Hidupkan komputer dengan menekan saklar pada komputer, jangan menghidupkan komputer dengan memasukkan colokan ke stop kontak ketika saklar dalam keadaan on.
- 4) Jalankan program Turbo Pascal.
- 5) Ujilah setiap contoh program pada uraian materi 3 di atas.
- 6) Buatlah program untuk membentuk tabel seperti di bawah ini:

```
A B C D E F G H I J K L M N O P
B C D E F G H I J K L M N O P
C D E F G H I J K L M N O P
D E F G H I J K L M N O P
E F G H I J K L M N O P
F G H I J K L M N O P
G H I J K L M N O P
H I J K L M N O P
I J K L M N O P
J K L M N O P
K L M N O P
L M N O P
M N O P
N O P
O P
P
```

- 7) Ujilah program anda !
- 8) Setelah selesai, laporkan hasil kerja Anda kepada instruktur.
- 9) Setelah diteliti matikan komputer dan rapikan seperti semula.

4. Kegiatan Belajar 4 : Pesan Kesalahan Kompilasi

a. Tujuan Kegiatan Pemelajaran

Setelah melaksanakan kegiatan pemelajaran 4 diharapkan peserta diklat dapat memahami pesan kesalahan kompilasi dan memperbaiki program.

b. Uraian Materi 4

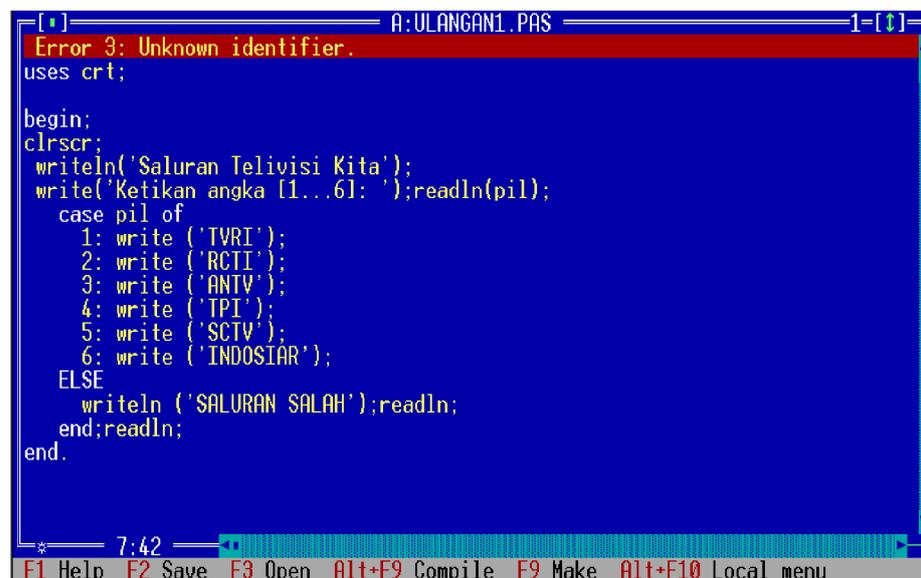
Pada pembuatan program dapat terjadi 2 buah kesalahan yaitu kesalahan sewaktu proses kompilasi dan kesalahan sewaktu program dijalankan. Pada pemrograman dengan Turbo Pascal apabila terjadi kesalahan maka akan ditunjukkan nomor kesalahan dan pesan kesalahannya.

Misalnya ada program yang masih mengandung kesalahan sebagai berikut:

```
program contoh;
uses crt;
begin;
  clrscr;
  writeln('Saluran Televisi Kita');
  write('Ketikan angka [1...6]: ');readln(pil);
  case pil of
    1: write ('TVRI');
    2: write ('RCTI');
    3: write ('ANTV');
    4: write ('TPI');
    5: write ('SCTV');
    6: write ('INDOSIAR');
  ELSE
    writeln ('SALURAN SALAH');readln;
  end;readln;
```

end.

Apabila program dikompilasi, akan terdapat pesan kesalahan (teks diblok warna merah) seperti di bawah ini :

A screenshot of the Turbo Pascal IDE showing a compilation error. The window title is 'A:ULANGAN1.PAS'. A red banner at the top displays the error message: 'Error 3: Unknown identifier.' Below this, the source code is visible, with the line 'write('Ketikan angka [1..6]: ');readln(pil);' highlighted. The code includes a 'case pil of' block with six options for TVRI, RCTI, ANTV, TPI, SCTV, and INDOSIAR, followed by an 'ELSE' clause that prints 'SALURAN SALAH'. The error occurs because the variable 'pil' has not been declared before being used in the 'write' statement. The status bar at the bottom shows the time as 7:42 and various function key shortcuts like F1 Help, F2 Save, F3 Open, Alt+F9 Compile, F9 Make, and Alt+F10 Local menu.

```

Error 3: Unknown identifier.
uses crt;

begin;
clrscr;
writeln('Saluran Telivisi Kita');
write('Ketikan angka [1..6]: ');readln(pil);
  case pil of
    1: write ('TVRI');
    2: write ('RCTI');
    3: write ('ANTV');
    4: write ('TPI');
    5: write ('SCTV');
    6: write ('INDOSIAR');
  ELSE
    writeln ('SALURAN SALAH');readln;
  end;readln;
end.
```

Gambar 2. Pesan Kesalahan pada Turbo Pascal

Pesan kesalahan akan ditampilkan urut dengan nomor kesalahan dan nama kesalahan. Pada Turbo Pascal apabila terjadi kesalahan kita akan langsung ditunjukkan tempat kita membuat kesalahan, seperti contoh di atas kita langsung ditunjukkan pada identifier pil. Pesan kesalahan yaitu Error 3 Unknown Identifier, hal tersebut berarti bahwa identifier pil belum dideklarasikan, untuk memperbaikinya kita harus mendeklarasikan identifier pil seperti di bawah ini:

```

program contoh;
uses crt;
variabel pil:integer;
begin;
  clrscr;
  writeln('Saluran Telivisi Kita');
```

```

write('Ketikan angka [1...6]: ');readln(pil);
  case pil of
    1: write ('TVRI');
    2: write ('RCTI');
    3: write ('ANTV');
    4: write ('TPI');
    5: write ('SCTV');
    6: write ('INDOSIAR');
  ELSE
    writeln ('SALURAN SALAH');readln;
  end;readln;
end.

```

Secara lengkap, nomor dan pesan kesalahan saat proses kompilasi pada Turbo Pascal adalah sebagai berikut:

1 Out of memory

Kesalahan ini terjadi bila memory tidak mencukupi sewaktu proses kompilasi.

2 Identifier expected

Suatu pengenalan (identifier) harus ada diposisi ini. Kesalahan ini mungkin

3 Unknown identifier

Pengenalan belum dideklarasikan

4 Duplicate identifier

Pengenalan telah digunakan dalam blok ini

5 Syntax error

Suatu karakter tidak sah telah ditemukan di program sumber. Kemungkinan kesalahan ini disebabkan lupa menuliskan tanda petik diantara suatu konstanta string.

6 Error in real constant

Konstanta tipe real salah

7 Error in integer constant

Konstanta tipe integer salah

8 String constant exceeds line

Konstanta string melebihi suatu baris

9 Too many nested files

Kesalahan ini disebabkan karena menggunakan include file yang bersarang lebih dari 5

10 Unexpected end of file

Kesalahan ini disebabkan karena program selesai tidak semestinya kemungkinan terjadi karena Begin dan End tidakimbang, suatu include file berakhir di tengah bagian pernyataan dan tidak menutup suatu komentar program.

11 Line too long

Maksimum panjang suatu baris adalah 126 karakter

12 Type identifier expected

Suatu pengenal tidak menunjukkan tipe yang seharusnya

13 Too many open files

Kesalahan ini adalah tidak dapat membukanya beberapa file, yang disebabkan oleh file CONFIG.SYS tidak ada atau jumlah file yang disebutkan di command CONFIG.SYS terlalu sedikit.

14 Invalid file name

Nama sari suatu file tidak benar atau mengandung jalur pencarian (path) yang tidak ada di disk

15 File not found

File tidak ditemukan di directory yang aktif

16 Disk full

Disk penuh, hapuslah file-file yang sudah tidak digunakan

17 Invalid compiler directive

Kesalahan pada compiler directive yang digunakan, disebabkan oleh kemungkinan huruf compile directive tidak dikenal, satu atau lebih parameter di compiler directive tidak benar dan penggunaan compiler directive yang bersifat global ketika kompilasi terhadap suatu badan program utama mulai dilakukan.

18 Too many files

Terlalu banyak file yang terlibat sewaktu kompilasi suatu program atau unit. Usahakanlah untuk tidak menggunakan terlalu banyak file, yaitu dengan cara menggabung beberapa include file.

19 Undefined type in pointer definition

Tipe ini sebelumnya telah dihubungkan dengan suatu deklarasi tipe pointer, tetapi belum pernah dideklarasikan.

20 Variable identifier expected

Pengenal ini tidak menunjukkan suatu variabel seperti yang seharusnya

21 Error in type

Simbol ini tidak dapat digunakan untuk memulai suatu deklarasi tipe

22 Structure too large

Ukuran maksimum yang diijinkan untuk tipe terstruktur adalah sebanyak 65520 byte

23 Set base type out of range

Base type dari suatu himpunan (set) harus berupa suatu subrange dengan batas jangkauan 0..255 atau suatu tipe enumerated yang tidak lebih dari 256 kemungkinan nilai.

24 File components may not be files

Konstruksi berupa file di dalam file tidak diijinkan

25 Invalid string length.

Panjang suatu string yang dideklarasikan harus diantara 1..255

26 Type mismatch

Kesalahan ini disebabkan karena tipe variabel dan ungkapan yang tidak sesuai di statement pengerjaan, tipe parameter

formal dan parameter nyata yang tidak sesuai pada penggunaan suatu prosedur atau fungsi, tipe ungkapan dan tipe indeks yang tidak sesuai untuk indeks suatu larik dan tipe operand yang tidak sesuai di suatu ungkapan.

27 Invalid subrange base type

Base type di suatu subrange yang tidak benar, harus berupa tipe ordinal

28 Lower bound greater than upper bound

Pendeklarasian suatu tipe subrange yang tidak benar, karena nilai batas bawahnya lebih besar dengan nilai batas atasnya.

29 Ordinal type expected

Tipe real, string, tipe terstruktur dan tipe pointer tidak diijinkan disini, harus berupa tipe ordinal.

30 Integer constant expected

Konstanta integer harus digunakan disini

31 Constant expected

Suatu konstanta harus digunakan disini

32 Integer or real constant expected

Konstanta integer atau real harus digunakan disini

33 Type identifier expected

Suatu pengenalan tidak menunjukkan suatu tipe yang seharusnya

34 Invalid function result type

Tipe hasil dari suatu fungsi tidak benar dan yang benar adalah semua tipe sederhana, tipe string atau tipe pointer

35 Label identifier expected

Suatu pengenalan tidak menunjukkan suatu label yang seharusnya

36 BEGIN expected

Kata cadangan Begin diharapkan

37 END expected

Kata cadangan End diharapkan

38 Integer expression expected

Ungkapan yang mengikuti harus bertipe integer

39 Ordinal expression expected

Ungkapan yang mengikuti harus bertipe ordinal

40 Boolean expression expected

Ungkapan yang mengikuti harus bertipe boolean

41 Operand types do not match operator

Operator yang digunakan tidak dapat untuk tipe operannya, misalnya operator DIV tidak dapat menggunakan operand tipe Char atau String

42 Error in expression

Kesalahan di dalam suatu ungkapan yang mungkin disebabkan Anda lupa meletakkan suatu operator diantara dua operand

43 Illegal assignment

Suatu pengerjaan yang salah, disebabkan oleh kemungkinan variabel file dan variabel tak bertipe tidak dapat diisi dengan suatu nilai

44 Field identifier expected

Suatu pengenalan tidak menunjukkan suatu field di variabel record yang mendahulinya

45 Object file too large

Turbo Pascal tidak dapat menghubungkan (link) file-file. OBJ yang lebih besar dari 64 K

46 Undefined external

Prosedur External atau fungsi External tidak mengandung definisi PUBLIC yang dapat ditemukan di file objek. Telitilah kembali file sumber .ASM.

47 Invalid object file record

File .OBJ mengandung record objek yang tidak benar. Yakinkanlah bahwa file ini adalah benar-benar file .OBJ

48 Code segment too large

Ukuran maksimum dari kode suatu program atau unit adalah 65520 byte

49 Data segment too large

Ukuran maksimum dari segment data suatu program 65520 byte, termasuk dengan data yang dideklarasikan di unit. Bila membutuhkan data yang bersifat global melebihi batas ini,

maka dapat dideklarasikan sebagai pointer dan alokasikanlah secara dinamik di heap menggunakan prosedur standar new.

50 DO expected

Statement DO diharapkan

51 Invalid PUBLIC definition

Kesalahan ini disebabkan karena kemungkinan suatu pengenalan dibuat public melalui suatu directive PUBLIC di bahasa perakit, tetapi tidak sesuai dengan deklarasi External di program Pascal atau unit, dua atau lebih directive PUBLIC di bahasa perakit mendefinisikan pengenalan yang sama dan file .OBJ mendefinisikan simbol PUBLIC yang tidak ada di segment CODE

52 Invalid EXTRN definition

Kesalahan ini disebabkan oleh kemungkinan suatu pengenalan dihubungkan lewat directive EXTRN pada bahasa perakit, tetapi tidak dideklarasikan di program Pascal atau unit, tidak juga dibagikan penghubung di suatu unit yang digunakan oleh program, yang kedua adalah suatu pengenalan sebagai variabel absolut dan yang terakhir disebabkan karena suatu pengenalan digunakan di prosedur atau fungsi Inline

53 Too many EXTRN definitions

Turbo Pascal tidak dapat menangani file .OBJ yang mempunyai lebih dari 256 definisi EXTRN

54 OF expected

Kata cadangan OF diharapkan

55 INTERFACE expected

Kata cadangan interface diharapkan

56 Invalid relocatable reference

Kesalahan ini disebabkan karena

- o Anda mencoba mendeklarasikan suatu variabel pada segment DATA di file .OBJ
- o File .OBJ mengandung byte-sized yang dihubungkan dengan simbol relocatable. Kesalahan ini terjadi bila Anda menggunakan operator HIGH dan LOW dengan simbol relocatable atau Anda hubungkan dengan simbol relocatable di directive DB
- o Suatu operand yang dihubungkan dengan simbol relocatable yang belum didefinisikan di segment CODE atau di segment DATA
- o Suatu operand yang dihubungkan dengan suatu prosedur atau fungsi EXTRN dengan suatu offset

57 THEN expected

Kata cadangan THEN diharapkan

58 TO or DOWNTO expected

Kata cadangan TO atau DOWN TO diharapkan

59 Undefined forward

Kesalahan ini disebabkan karena:

- Suatu prosedur atau fungsi telah dideklarasikan di bagian penghubung dari suatu unit, tetapi definisinya tidak tampak di bagian penerapan
- Suatu prosedur atau fungsi telah dideklarasikan dengan acuan Forward tetapi defisinya tidak ditemukan

60 Too many procedures

Turbo Pascal tidak mengijinkan lebih dari 512 prosedur atau fungsi tiap modulnya. Bila Anda mengkompilasi suatu program, rubahlah bebarapa prosedur atau fungsi menjadi satu unit. Bila anda mengkompilasi suatu unit, pecahlah menjadi dua atau lebih unit.

61 Invalid typecast

Kesalahan ini disebabkan karena :

- Ukuran dari acuan variabel dan tipe tujuannya berbeda di suatu variabel typecast
- Anda mencoba untuk melakukan typecast suatu ungkapan yang hanya sebuah acuan variabel saja yang diperbolehkan

62 Devision by zero

Operand sebelumnya terbagi dengan nilai nol

63 Invalid file type

Tipe dari file tidak didukung oleh prosedur penanganan file, misalnya melakukan prosedur standar Seek atau file teks

64 Cannot Read or Write variables of this type

Kesalahan ini disebabkan oleh:

- Prosedur standar Read atau Readln mencoba membaca data yang bukan tipe Char, Integer, Real atau String
- Prosedur standar Write atau Writeln mencoba menampilkan data yang bukan tipe Char, Integer, Real, String atau Boolean

65 Pointer variable expected

Variabel sebelumnya harus bertipe Pointer

66 String variable expected

Variabel sebelumnya harus bertipe String

67 String expression expected

Ungkapan sebelumnya harus bertipe String

68 Circular unit reference

Dua unit tidak diijinkan untuk menggunakan satu dengan yang lainnya

69 Unit name mismatch

Nama dari unit di file .TPU tidak sama dengan nama dari unit yang disebutkan di Uses

70 Unit version mismatch

Satu atau lebih unit yang digunakan oleh unit ini telah berubah sejak terakhir kali unit di kompilasi. Gunakanlah Compile/Make atau Compile/Build di integrated environment atau option /M atau /B di command-line compiler untuk secara otomatis mengkompilasi unit-unit yang membutuhkan kompilasi ulang.

71 Duplicate unit name

Anda telah menggunakan nama unit ini sebelumnya di Uses, sehingga ada nama unit yang kembar

72 Unit file format error

Terdapat kesalahan di file .TPU. yakinkanlah bahwa file tersebut adalah file unit

73 Implementation expected

Kata cadangan Implementation diharapkan

74 Constant and case types do not match

Tipe dari konstanta Case tidak sesuai dengan ungkapan selector di statemen Case

75 Record variable expected

Variabel sebelumnya harus bertipe record

76 Constant out of range

Kealahan ini disebabkan karena Anda mencoba mengindeks suatu larik dengan konstanta yang diluar jangkauannya, anda mencoba untuk memasukkan nilai yang diluar jangkauan suatu variabel dan anda mencoba untuk mengirimkan nilai konstanta yang diluar jangkauan ke suatu prosedur atau fungsi

77 File variable expected

Variabel sebelumnya harus berupa tipe file

78 Pointer expression expected

Ungkapan sebelumnya harus berupa tipe Pointer

- 79** Integer atau real expression expected
Ungkapan sebelumnya harus berupa tipe Integer atau Real
- 80** Label not within current block
Suatu statement Goto tidak dapat menunjukkan suatu label yang berada di luar blok
- 81** Label already defined
Suatu label telah pernah digunakan sebelumnya
- 82** Undefined label in preceding statement part
Suatu label telah dideklarasikan dan dihubungkan di bagian pernyataan sebelumnya, tetapi tidak pernah didefinisikan
- 83** Invalid @ argument.
Argumen dari operator @ tidak benar dan yang benar adalah berupa acuan variabel atau pengenal prosedur atau pengenal fungsi
- 84** UNIT expected
Kata cadangan Unit diharapkan
- 85** ";" expected
Simbol ";" diharapkan
- 86** ":" expected
Simbol ":" diharapkan
- 87** "," expected
Simbol "," diharapkan

88 "(" expected

Simbol "(" diharapkan

89 ")" expected

Simbol ")" diharapkan

90 "=" expected

Simbol "=" diharapkan

91 ":=" expected

Simbol ":=" diharapkan

92 "[" or "(" expected

Simbol "[" atau "(" Diharapkan

93 "[" or ".)" expected

Simbol "[" atau ".)" diharapkan

94 "." expected

Simbol "." diharapkan

95 ".." expected

Simbol ".." diharapkan

96 Too many variables

Kesalahan ini disebabkan karena ukuran total dari variabel global yang dideklarasikan di suatu program atau unit tidak melebihi 64K dan ukuran total dari variabel lokal dideklarasikan di suatu program atau unit tidak dapat melebihi 64 K

97 Invalid FOR control variable

Variabel kontrol di statement FOR harus berupa variabel tipe sederhana yang dideklarasikan di bagian deklarasi

98 Integer variable expected

Variabel sebelumnya harus berupa tipe integer

99 Files are not allowed here

Suatu konstanta tidak dapat digunakan sebagai tipe dari suatu file

100 String length mismatch

Panjang dari konstanta string tidak sesuai dengan jumlah komponen dari karik karakternya

101 Invalid ordering of fields

Field dari suatu konstanta tipe record harus ditulis dengan urutan yang sma deklarasinya

102 String constant expected

Konstanta string diharapkan

103 Integer or real variable expected

Variabel sebelumnya harus berupa tipe integer atau real

104 Ordinal variable expected

Variabel sebelumnya harus berupa tipe ordinal

105 INLINE error

Operator < tidak diijinkan di relocatable references yang dihubungkan dengan suatu variabel, acuan ini harus selalu word-sized

106 Character expression expected

Ungkapan karakter diharapkan

107 Too many relocation items

Jumlah dari tabel penampung di file .EXE melebihi 64 K yang merupakan batas dari Turbo Pascal. Jika anda menemui kesalahan ini, berarti program Anda terlalu besar bagi linker Turbo Pascal untuk menanganinya. Juga mungkin program Anda terlalu besar dieksekusi oleh DOS

108 Not enough memory to run program

Kesalahan ini karena memori yang digunakan tidak cukup untuk menjalankan suatu program lewat integrated environment. Kalau anda menggunakan program-program resident, hapuslah program ini dan bila memori masih tidak cukup, kompilasi program ke disk dan jalankan di prompt DOS

109 Cannot find EXE file

Kesalahan ini mungkin disebabkan file .EXE yang telah dihasilkan oleh kompiler hilang dari disk

110 Cannot run a unit

Anda tidak dapat menjalankan secara langsung suatu unit, harus dijalankan suatu program.

111 Compilation aborted

Proses kompilasi dihentikan dengan penekanan tombol Ctrl-Break

112 CASE constant out of range

Untuk statement Case tipe Integer, konstanta harus bernilai diantara -32768..32767

113 Error in statement

Suatu statement diawali dengan simbol yang tidak benar

114 Cannot call an interrupt procedure

Anda tidak dapat secara langsung memanggil suatu prosedur interrupt

115 Must have an 8087 to compile this

Kompiler membutuhkan suatu coprocessor 8087 untuk mengkompilasi suatu program atau unit bila digunakan directive {\$N+}

116 Must be 8087 mode to compile this

Anda menggunakan tipe data real 8087, yaitu Single, Double, Extended dan Comp yang membutuhkan coprocessor 8087 dengan status directive {\$N+}

117 Target address not found

Command compile/ find error di integreted environment atau option /F di command line tidak dapat menemukan suatu statement di suatu alamat tertentu

118 Include files are not allowed here

Setiap bagian pernyataan harus berada seluruhnya di satu file

119 TPM file format error

File .TPM tidak benar, yakinlah bahwa file ini benar-benar file .TPM

120 NIL expected

Kata cadangan NIL diharapkan

121 Invalid qualifier

Kesalahan ini disebabkan karena anda mencoba mengindeks suatu variabel yang bukan suatu lirik, anda mencoba menentukan suatu field di suatu variabel yang bukan suatu record dan anda mencoba untuk mengacu suatu variabel yang bukan suatu pointer.

122 Invalid variable reference

Bentuk sebelumnya diikuti dengan suatu sintaks dari variabel acuan, tetapi tidak menunjukkan suatu lokasi memori.

123 Too many symbols

Suatu program atau unit menggunakan simbol yang dideklarasikan lebih dari 64 K

124 Statement part too large

Turbo Pascal membatasi ukuran dari bagian pernyataan sekitar 24K. Jika Anda menemui kesalahan ini pindahkan bagian ini ke dalam satu atau lebih prosedur.

125 Module has no debug information

Suatu runtime error terjadi di suatu modul (program atau unit) yang tidak mempunyai informasi pelacakan

126 Files must be var parameters

Kesalahan ini disebabkan karena Anda berusaha untuk mendeklarasikan parameter nilai di tipe file. Parameter di tipe file harus berupa parameter var.

127 Too many conditional symbols

Sudah tidak tersedia tempat untuk mendefinisikan conditional symbols lebih lanjut. Cobalah menghilangkan beberapa simbol atau memendekkan nama dari simbol.

128 Misplaced conditional directive

Kompiler menemukan suatu directive `{$ELSE}` atau `{$ENDIF}` yang tidak mempunyai pasangan directive `{$IFDEF}`, `{$IFNDEF}` atau `{$IFOPT}`

129 ENDIF directive missing

File sumber berakhir di dalam suatu konstruksi kompilasi bersyarat (conditional compilation) yang disebabkan tidak samanya jumlah directive `{$Ifxxx}` dengan `{$ENDIF}`

130 Error in initial conditional defines

Simbol bersyarat awal yang disebutkan di Options/Compiler/Conditional defines atau directive `/D` tidak benar.

131 Header does not match previous definition

Kesalahan ini disebabkan oleh kemungkinan judul dari prosedur atau fungsi yang disebutkan di bagian penghubung tidak sama dengan judul ini dan judul dari prosedur atau fungsi yang disebutkan di deklarasi Forward tidak sama dengan judul ini.

132 Critical disk error

Suatu kesalahan kritis terjadi sewaktu proses kompilasi, misalnya drive tidak siap

133 Old map file

File .TPM merupakan file yang lama dibandingkan dengan file .EXE bersangkutan. Ini merupakan indikasi bahwa terakhir kali Anda mengkompilasi program, suatu file .TPM tidak dihasilkan

c. Rangkuman 4

Turbo Pascal saat di kompilasi akan menunjukkan kesalahan urut dari kesalahan di awal program sampai akhir program. Apabila kesalahan pertama telah diperbaiki maka Turbo Pascal akan mencari kesalahan di bawahnya.

d. Tugas 4

- 1) Pelajarilah uraian materi tentang Pesan Kesalahan Kompilasi !
- 2) Bagaimanakah caranya memperbaiki apabila dikompilasi terdapat pesan kesalahan Division by zero!

e. Tes Formatif 4

- 1) Sebutkan masing-masing arti dari pesan kesalahan berikut ini:
 - a) Duplicate Identifier

b) Unexpected end of file

c) Division by zero

2) Perbaikilah program berikut ini :

```
Const V = 'vokal';
      K = 'konsonan';
      T = 'tanda baca';
      A = 'angka';

Var kata : String;
    jawab : Char;

Begin
ClrScr;
Writeln('Program menghitung jumlah karakter kata/kalimat');
Writeln('Misalnya :');
Writeln('Jumlah karakter ',V,' adalah ',length(V),'
karakter');
Writeln('Jumlah karakter ',K,' adalah ',length(K),'
karakter');
    Writeln('Jumlah karakter ',T,' adalah ',length(T),'
karakter');
    Writeln('Jumlah karakter ',A,' adalah ',length(A),'
karakter');
Writeln;
jawab := 'Y';
Writeln('Apakah Anda ingin menghitung jumlah karakter ?');
Writeln('(Tekan "Y" jika ya atau "T" jika tidak)');
Readln(jawab);
While (jawab = 'Y') or (jawab = 'y') do
Begin
    Write('Masukkan kata-kata Anda : '); Readln(kata);
```

```

        WriteLn('Jumlah karakter ',kata,' adalah
        ',length(kata),' karakter');
        WriteLn;
        Write('Masih ingin menghitung karakter lagi ? ');
        ReadLn(jawab);
    End;
End;
End;

```

f. Kunci Jawaban Formatif 4

1) Arti dari masing-masing pesan Kesalahan adalah sebagai berikut:

- a) Duplicate Identifier : Pengenal telah digunakan dalam blok ini
- b) Unexpected end of file : Kesalahan ini disebabkan karena program selesai tidak semestinya kemungkinan terjadi karena Begin dan End tidakimbang, suatu include file berakhir di tengah bagian pernyataan dan tidak menutup suatu komentar program
- c) Division by zero : Operand sebelumnya terbagi dengan nilai nol

2) Program yang betul:

```

Uses Crt;
Const V = 'vokal';
      K = 'konsonan';
      T = 'tanda baca';
      A = 'angka';

```

```

Var kata : String;
    jawab : Char;
Begin
    ClrScr;
    Writeln('Program menghitung jumlah karakter
kata/kalimat');
    Writeln('Misalnya :');
    Writeln('Jumlah karakter ',V,' adalah ',length(V),'
karakter');
    Writeln('Jumlah karakter ',K,' adalah ',length(K),'
karakter');
    Writeln('Jumlah karakter ',T,' adalah ',length(T),'
karakter');
    Writeln('Jumlah karakter ',A,' adalah ',length(A),'
karakter');
Writeln;
Begin
    jawab := 'Y';
    Writeln('Apakah Anda ingin menghitung jumlah karakter
?');
    Writeln('(Tekan "Y" jika ya atau "T" jika tidak)');
    Readln(jawab);
    While (jawab = 'Y') or (jawab = 'y') do
    Begin
        Write('Masukkan kata-kata Anda : '); Readln(kata);
        Writeln('Jumlah karakter ',kata,' adalah
',length(kata),' karakter');
    Writeln;

```

```
        Write('Masih ingin menghitung karakter lagi ? ');
        Readln(jawab);
    End;
End;
End.
```

g. Lembar Kerja 4

Alat dan Bahan

PC yang sudah terinstal Turbo Pascal 7.0

Kesehatan dan Keselamatan Kerja

- 1) Berdo'alah sebelum memulai kegiatan belajar.
- 2) Bacalah dan pahami petunjuk praktikum pada setiap lembar kegiatan belajar.
- 3) Pastikan komputer dalam keadaan baik, semua kabel penghubung terhubung dengan benar.
- 4) Jangan meletakkan benda yang dapat mengeluarkan medan elektromagnetik di dekat komputer (magnet, handphone, dan sebagainya).
- 5) Gunakanlah komputer sesuai fungsinya dengan hati-hati, jangan bermain game.
- 6) Setelah selesai, matikan komputer dengan benar!

Langkah Kerja

- 1) Siapkanlah peralatan yang akan digunakan!
- 2) Periksa semua kabel penghubung pada komputer.
- 3) Hidupkan komputer dengan menekan saklar pada komputer, jangan menghidupkan komputer dengan memasukkan colokan ke stop kontak ketika saklar dalam keadaan on.
- 4) Jalankan program Turbo Pascal.

- 5) Tulislah contoh program pada Uraian Materi 4 kemudian kompilasikan dan perbaiki kesalahannya.
- 6) Buatlah program di bawah ini:

```
Uses Crt;
Var x, tebak, jum : char;
Begin
  Randomize;
  jum := 0;
  While (jum <= 10)
  Begin
    ClrScr;
    Writeln('Komputer akan menampilkan 1 angka
    random');
    Writeln('yang nilainya berkisar antara 0 s/d 999. ');
    Writeln('Anda mempunyai 10 kesempatan untuk
    menebaknya. ');
    Writeln('Silahkan menebak ... ');
    x := Random(1000);
    jum := 0;
    tebak := -1;
    While (tebak <> x) do
    Begin
      tebak := 0;
      Writeln('Tebakan Anda ? ');
      Readln(tebak);
      If (tebak < x) then
      Begin
        Writeln(tebak, 'Nilai tebakan Anda terlalu kecil');
      End;
```

```

    (tebak > x) then
    Begin
        Writeln(tebak,'Nilai tebakan Anda terlalu besar');
    End;
    jum := jum + 1;
End;
If (tebak = x) then
Begin
    Writeln(tebak,'Anda memang beruntung');
End;
If (jum > 10) then
Begin
    Writeln('Kegagalan adalah sukses yang tertunda ...');
    Writeln('Jangan pikirkan hari esok jika anda belum
        makan hari ini');
End;
End;
Readln;
End.

```

- 7) Kompilasi program tersebut dan perbaikilah!
- 8) Setelah selesai, laporkan hasil kerja Anda kepada instruktur.
- 9) Setelah diteliti matikan komputer dan rapikan seperti semula.

BAB III EVALUASI

A. PERTANYAAN

1. Sebutkan macam-macam bentuk operator matematika dan relasi dan terangkan secara singkat artinya !
2. Sebutkan macam-macam bentuk perulangan?
3. Buatlah program Luas Persegi Panjang, untuk memasukan dua buah bilangan dari keyboard, yang kemudian akan ditampilkan hasil dari Luas Persegi Panjang ?
4. Buatlah program Luas Segitiga, untuk memasukan dua buah bilangan dari keyboard, yang kemudian akan ditampilkan hasil dari Segitiga ?

B. KUNCI JAWABAN EVALUASI

1. Macam- macam operator matematika:

Operator	Arti	Prioritas
*	Kali	1
/	Bagi (real)	1
DIV	Bagi (bulat)	1
MOD	Sisa pembagian	1
+	Tambah	2
-	Kurang	2

Macam-macam operator relasi:

Operator	Keterangan
>	Lebih dari
<	Kurang dari
>=	Lebih dari atau sama dengan
<=	Kurang dari atau sama dengan
<>	Tidak sama dengan
=	Sama dengan
IN	Perbandingan suatu nilai dengan suatu himpunan

2. Macam- macam bentuk perulangan :

- a. FOR
- b. WHILE
- c. UNTIL

3. Program:

```
program luas_persegipanjang;  
uses crt;  
var a,b,c :integer;  
begin  
  write('Panjang persegi panjang : ');readln(a);  
  write('Lebar persegi Panjang : ');readln(b);  
  c:=a*b;  
  writeln('Luas persegi panjang : ',c);readln;  
end.
```

4. Program

```
program Luas_segitiga;  
uses crt;  
var a,b,c :integer;  
begin  
  write('Alas Segitiga      : ');readln(a);  
  write('Tinggi Segetiga   : ');readln(b);  
  c:=0.5(a*b);  
  writeln('Luas segitiga: ',c);readln;  
end.
```

C. KRITERIA PENILAIAN

Aspek	Skor (1-10)	Bobot	Nilai	Keterangan
Kognitif (soal no 1 s/d 2)		2		Syarat lulus nilai minimal 70 dan skor setiap aspek minimal 7
Kebenaran program		6		
Kerapian penulisan program		1		
Ketepatan waktu		0.5		
Ketepatan penggunaan program Turbo Pascal		0.5		
Nilai Akhir				

Kategori kelulusan:

70 – 79 : Memenuhi kriteria minimal. Dapat bekerja dengan bimbingan.

80 – 89 : Memenuhi kriteria minimal. Dapat bekerja tanpa bimbingan.

90 – 100 : Di atas kriteria minimal. Dapat bekerja tanpa bimbingan.

BAB IV

PENUTUP

Setelah menyelesaikan modul ini dan mengerjakan semua tugas serta evaluasi maka berdasarkan kriteria penilaian, peserta diklat dapat dinyatakan lulus/ tidak lulus. Apabila dinyatakan lulus maka dapat melanjutkan ke modul berikutnya sesuai dengan alur peta kedudukan modul, sedangkan apabila dinyatakan tidak lulus maka peserta diklat harus mengulang modul ini dan tidak diperkenankan mengambil modul selanjutnya.

DAFTAR PUSTAKA

Abdul Kadir, (1999). Pemrograman Pascal Buku I. Andi Offset, Yogyakarta.

Abdul Kadir, (2002). Pemrograman Pascal Buku II. Andi Offset,
Yogyakarta.

Jogiyanto, (2001). Turbo Pascal Versi 5.0 Jilid 1. Andi Offset, Yogyakarta.

Jogiyanto, (1989). Turbo Pascal Versi 5.0 Jilid 2. Andi Offset, Yogyakarta.